ABSTRACT

Network security has gained significant attention in research and industrial communities. Due to the increasing threat of the network intrusion, firewalls have become important elements of the security policy. Firewall performance highly depends toward number of rules, because the large more rules the consequence makes downhill performance progressively. Firewall can be allow or deny access network packets incoming and outgoing into Local Area Network (LAN), but firewall can not detect intrusion. To distinguishing an intrusion network packet or normal is very difficult and takes a lot of time. An analyst must review all the network traffics previously. In this study, a new way to make the rules that can determine network packet is intrusion or normal automatically. These rules implemented into firewall as prevention, which if there is a network packet that match these rules then network packet will be dropped. This is called Network Intrusion Prevention System (NIPS). These rules are generated based on Network Intrusion Detection System (NIDS) and Iterative Dichotomiser 3 (ID3) Algorithm Decision Tree Classifier, which as data training is intrusion network packet and normal network packets from previous network traffics. The experiment is successful, which can generate the rules then implemented into a firewall and drop the intrusion network packet automatically. Moreover, this way can minimize number of rules in firewall.

ABSTRAK

Keselamatan rangkaian telah mendapat perhatian penting dalam penyelidikan dan masyarakat industri. Disebabkan peningkatan ancaman gangguan rangkaian, firewall telah menjadi elemen penting bagi polisi keselamatan. Kejayaan firewall sangat bergantung terhadap jumlah peraturan, kerana peraturan-peraturan yang lebih besar mengakibatkan prestasi semakin menurun. Firewall boleh membenar atau menolak paket akses rangkaian yang masuk dan keluar daripada Local Area Network (LAN), tetapi firewall tidak dapat mengesan intrusi. Untuk membezakan pakej intrusi rangkaian atau normal adalah sangat sukar dan memerlukan banyak masa. Seseorang penganalisis perlu memeriksa semua rangkaian trafik pada masa sebelumnya. Dalam kajian ini, suatu cara baru untuk membuat peraturan yang boleh menentukan pakej rangkaian intrusi atau normal secara automatik. Peraturan-peraturan ini diimplementasikan ke dalam firewall sebagai pencegahan, yang mana sekiranya ada pakej rangkaian yang sesuai dengan peraturan-peraturan ini pakej rangkaian akan diabaikan. Ini disebut dengan Network Intrusion Prevention System (NIPS). Peraturan-peraturan ini dijanakan berdasarkan Network Intrusion Detection System (NIDS) dan Iteratif Dichotomiser 3 (ID3) Algorithm Decision Tree Classifier, yang mana data latihan adalah intrusi pakej rangkaian dan normal pakej rangkaian daripada lalu lintas rangkaian terdahulu. Eksperimen ini telah berjaya, yang dapat menghasilkan peraturan-peraturan yang diimplementasikan ke dalam firewall dan pakej intrusi rangkaian diabaikan secara automatik. Selain itu, cara ini dapat meminimumkan jumlah peraturan dalam firewall.

TABLE OF CONTENTS

CHAPTER 5

CONCLUSION AND RECOMENDATIONS

5.1 CONCLUSION

Network security has become a critical issue with the development of business and other transactions through the internet system. Firewall is one important element in network security systems because it can drop the a network packet incoming to LAN. Firewall can not define a network packet is intrusion or normal. An intrusion can be defined as any set of actions that threaten the integrity, confidentiality or availability of a network resource, such as user account, file system, system kernels and so on. Specifies the network packet is intrusion to be implemented on the firewall rules are very difficult. An analyst should reviews large data from network traffics previously. Meanwhile, to update and manage the firewall rules are very difficult and takes a lot of time.

By using the ID3 algorithm decision tree classifier to generate rules where network traffics as data training. A network packet from the network traffics can be seen from the log files. To determine the network packet is intrusion or normal using *snort* application.

Rules generated can determine a new network packet is intrusion or normal. These rules are implemented into firewall automatically where the firewall will drop the network packet that match those rules as intrusion.

This research contributes: first, to create rules that function to determine the network traffic is the intrusion and then implemented rules into the firewall as a prevention automatically. This combination is called NIPS, because it can determine network packet is intrusion automatically and can be prevented by using a firewall.

This method can minimize the number of rules in the firewall, where one rule can replace two or more rules and make a better firewall performance. This is very helpful and easier to update and manage the firewall rules.

This research made software that is named *nips-nids2s3*, this software helps construct decision tree to generate the rules and implemented into firewall *iptables* automatically as prevention.

5.2 RECOMMENDATIONS

Data collection for intrusion do if the network packets do intrusion several times from the same source IP address. If intrusion is conducted just one time, it can be an unintentionally and ignored. Intruder do many times intrusion many times, with goal to find and get as much as information from a machine target. Intruder takes a lot of time and in many ways to get information such as port scans, ping, send packages, etc. It is impossible intruder to do intrusion just once.

In large computer networks, this condition will produces large data set and also generate so many rules. It is recommended reduces some rules become one rule using the port range or multi port and IP range or IP network by masking the same protocol and action.

For future work, the whole way this is done with real time system and performed on large computer networks such as Wide Area Network (WAN) and internet.

CHAPTER 1

INTRODUCTION

Several research articles have been published regarding firewall as prevention. Among them were those by Gollman D. (2006), Al-Shaer E.S. and Hamed H.H. (2004), Golnabi K. et al. (2006), Terpstra J.H. et al. (2004), Joko Y. and Onno W.P. (2008), Benelbahri M.A. and Bouhoula A. (2007), Tibbs R.W. and Oakes E.B. (2006), Winding R. et al. (2006), Suehring S. and Ziegler R.L. (2006), Katić T. and Pale P. (2007), Wenhui C. et al. (2006). Those articles revealed that updating firewall policy rules one of the current issues that still unsolved problems, where one of the problem is how to denied network packet intrusion (Gollman D., 2006; Al-Shaer E.S. and Hamed H.H., 2004; Golnabi K. et al., 2006; Tibbs R.W. and Oakes E.B., 2006; Suehring S. and Ziegler R.L., 2006 and Katić T. and Pale P., 2007). This is because several problems hinder in finding and creating the effective firewall rules of intrusion. Therefore, the study on this basis is initiated.

Development of the internet and the availability of tools for intrusion by hackers become critical component of network administration. An intrusion can be defined as actions that threaten the integrity, confidentiality or availability of a network resources (SANS Institute, 2008), such as user account, file system, system kernels and so on (Chandrasekar A. et al., 2009 and SANS Institute, 2008.)

Network security system is described by a firewall (Gollmann D., 2006; Al-Shaer E.S. and Hamed H.H., 2004, 2004; Guan X. and Yun-jie L, 2010; Golnabi K. et al., 2006; Tibbs R.W. and Oakes E.B., 2006; Suehring S. and Ziegler R.L., 2006; Katić T. and Pale P., 2007). Firewalls can limit certain network packet, but firewall cannot recognize that is network packet is an intrusion or attack. Conversely, there is no

Intrusion Detection System (IDS) software who can be immediately implemented into the firewall policy rules. IDS software can only generate an alarm or log such as *snort*, *honeypot* and *portsentry*.

1.1 NETWORK SECURITY AND INTRUSION DETECTION

Network Security arise from local computer network connected to wide-area network such as internet. During local network computer not connected to wide-area network, problem of network security is not be important. Network security explains the possibilities to arise from connected local network computer to wide-area network (Joko Y. and Onno W.P., 2008). The global internet connection, network security has gained significant attention in research and industrial communities. Due to the increasing threat of network attacks, firewalls have become important elements of the security policy is generally (Al-Shaer E.S. and Hamed H.H., 2004; Benelbahri M.A. and Bouhoula A., 2007 and Suehring S. and Ziegler R.L., 2006).

Firewall is network security and first line of defense against external network attacks and threats. Firewall controls or governs network access by allowing, denying or forward the incoming and outgoing network traffics (Guan X. and Yun-jie L., 2010; Golnabi K. et al., 2006; PC Perspective, 2008; Tibbs R.W. and Oakes E.B., 2006; Suehring S. and Ziegler R.L., 2006; Mitra S. and Acharya T., 2003; Katić T. and Pale P., 2007 and Yan Y., 2010).

Firewalls can protect network systems and minimize the risk of attacks to the network. Intrusion Detection System (IDS) is good for detecting the existence of intrusion. The technology is joining ability of IDS and protection, so-called Intrusion Prevention System (IPS) (Wiliam W.S.C., 2005). IPS can detect intrusion and then drop of intrusion as prevention. Intrusion prevention is an evolution of intrusion detection.

1.2 PROBLEM STATEMENT

Behavior of intrusion conducts an attack is slowly, continuous and requires a long time. They made preparations to find all information of the system, such as machine and operating system (OS) used, ports, administrator, software application used, network topology and so on. Intrusion learn from the system and perform an attack that no longer generate an alarm (Gollmann D., 2006). All of activities can be seen on network traffic logs. Log can provide a useful information and crucial to be able to distinguish normal activities and intrusive activities (Terpstra J.H. et al., 2004).

Network traffics can be observed from log files (Golnabi K. et al., 2006) as human pattern recognize (Winding R. et al., 2006). For illustration, in Table 1.1 there are 15 records network traffic which consists 10 intrusion network pakckets and 5 normal network packet. All the network packet of intrusion should be dropped into firewall rules for network security. In Figure 1.1 shows intrusion network packet in Table 1.1 that was implemented into firewall rules.

Table 1.1 Network traffics illustration

No	Source IP	Dest IP	Dest Port	Protocol	Intrusion
1	122.206.13.100	10.10.1.2	22	TCP	Yes
2	122.306.13.100	10.10.1.2	22	TCP	Yes
3	122.306.13.100	10.10.1.2	22	TCP	Yes
4	122.306.13.100	10.10.1.5	22	TCP	Yes
5	122.306.13.100	10.10.1.5	80	TCP	Yes
6	122.306.13.100	10.10.1.3	80	TCP	Yes
7	203.130.14.20	10.10.1.5	22	TCP	No
8	203.130.14.20	10.10.1.5	22	TCP	No
9	203.130.14.20	10.10.1.3	22	TCP	No
10	203.130.14.20	10.10.1.3	80	TCP	Yes
11	206.145.206.4	10.10.1.2	21	TCP	Yes
12	206.145.206.4	10.10.1.2	80	TCP	Yes
13	206.145.206.4	10.10.1.3	80	TCP	No
14	206.145.206.4	10.10.1.5	80	TCP	Yes
15	206.145.206.4	10.10.1.5	80	TCP	No

```
R1  : -A FORWARD -p tcp -s 122.206.13.100 –sport 1360  -d 10.10.1.2 --dport 22 -j DROP
R2  : -A FORWARD -p tcp -s 122.206.13.100 –sport 1425 -d 10.10.1.2 --dport 22 -j DROP
R3  : -A FORWARD -p tcp -s 122.206.13.100 –sport 1488 -d 10.10.1.2 --dport 22 -j DROP
R4  : -A FORWARD -p tcp -s 122.206.13.100 –sport 1559 -d 10.10.1.5 --dport 22 -j DROP
R5  : -A FORWARD -p tcp -s 122.206.13.100 –sport 1620 -d 10.10.1.5 --dport 80 -j DROP
R6  : -A FORWARD -p tcp -s 122.206.13.100 –sport 136 -d 10.10.1.3 --dport 22 -j DROP
R7  : -A FORWARD -p tcp -s 203.130.14.20 –sport 4607-d 10.10.1.3 --dport 80 -j DROP
R8  : -A FORWARD -p tcp -s 206.145.206.4 –sport 4690 -d 10.10.1.2 --dport 21 -j DROP
R9  : -A FORWARD -p tcp -s 206.145.206.4 –sport 1552 -d 10.10.1.2 --dport 80 -j DROP
R10 : -A FORWARD -p tcp -s 206.145.206.4 –sport 1330 -d 10.10.1.5 --dport 80 -j DROP
```

Figure 1.1 Firewall rules drop the intrusion network packet

Packet filtering firewall can limit the access to the connection base on parameters including protocol, source IP, destination IP, source port, destination port etc (Al-Shaer E.S. and Hamed H.H., 2004; Golnabi K. et al., 2006; Tibbs R.W. and Oakes E.B., 2006; Suehring S. and Ziegler R.L., 2006 and Katić T. and Pale P. , 2007). Packet filtering firewall has the character of the static thus function also static and has limitation (Suehring S. and Ziegler R.L., 2006 and Wiliam W.S.C., 2005). For example, access to the web server using port 80 allowed by the firewall policy rule, so from anywhere activities pass port 80 is allowed although there is attempt penetration by intruder. Therefore, these rules are in a constant need of updating by inserting, modifying or removing, tuning and validating (Al-Shaer E.S. and Hamed H.H., 2004; Golnabi K. et al., 2006; Suehring S. and Ziegler R.L., 2006 and Katić T. and Pale P. , 2007).

If every intrusion will be implemented to firewall rules which not carefully ordered and selective, this condition will makes the policy contains a large number of firewall rules. The possibility of policy anomaly will be happened relatively high, such as writing conflicting or redundant rules (Al-Shaer E.S. and Hamed H.H., 2004). This condition causes the performance of firewall decreases (Gollmann D., 2006; Dunham and Margareth H., 2002 and Benelbahri M.A. and Bouhoula A., 2007), because incoming and outgoing every network packet must be checked against the rules until the

rules found matching (Gollmann D., 2006; Golnabi K. et al. 2006 and Khalil R.K. et al., 2010).

The distinguishing an intrusion network packet manually through log files is difficult, because requiring a lot of time and tedious (Al-Shaer E.S. and Hamed H.H., 2004 and Golnabi K. et al., 2006). An analyst must review all of network traffic. For example in Table 1.1, that describe line 11, 12, 13, 14 and 15 with same source IP 206.145.206.4 were have two categories: intrusion activities and normal activities. It is required a method to create intrusion rules that be able to select a IP address which have two categories. Therefore, the necessary means to selectively and automatically determining a network packet is an intrusion that called NIDS and then to implemented into firewall rules as prevention. Combination both of NIDS and firewall namely NIPS.

The task of manually-manage firewall policy rules becomes very difficult and takes a lot of time, because the number of network traffics are increase and continuous. This huge task which requires a way to automatically update the firewall rules (Golnabi K. et al., 2006).

1.3 OBJECTIVES OF THE RESEARCH

The objectives of the research are as follows:

i. To propose a new framework of Network Intrusion Prevention System (NIPS) based on Network Intrusion Detection System (NIDS) and ID3 Algorithm Decision Tree Classifier.
ii. To implement and analyse the performance of the framework in the computer network environment.

1.4 OVERVIEW OF THE THESIS

The distinguish network traffics is intrusion or normal is very difficult and takes a lot of time. An analyst must review all the data of network traffics to find the intrusion then implemented into firewall rules as prevention.

This research makes rules that can determine network packet is intrusion, then rules are implemented into the firewall as NIPS. Rules resulting from the construct decision tree using ID3 algorithm and network packets as data training. Network packet obtained from the log files that record the activity of network traffics. Network packet intrusion and normal as data training extracted into five attributes: source IP address, destination IP address, source port, destination port and protocol. To determine the packet is a network intrusion or a normal use the *Snort* NIDS software. *Snort* also generate log files and parse log files that perform intrusion and normal.

1.4 ORGANIZATION OF THE THESIS

This thesis is organized as follows: Chapter 2 reviews the Intruder, Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Log Files, Firewall, Data Mining and Decision Tree Classifier of Data Mining. Chapter 3, proposes NIPS based on NIDS and ID3 algorithm decision tree classifier. Chapter 4, the implementation and analysis the performance rules in computer network environment. Finally, the Conclusion and Recommendations are presented in Chapter 5.

1.5 CONCLUSION

This chapter explains network security that gained significant attention in computer network. To distinguish the intrusion activities from normal activities of the network traffics is very difficult and require a lot of time. It is needed a method to define intrusion from network traffics automatically that called NIDS. Firewall able to protect network system and can minimization risk of intrusion. NIDS and firewall have become important element of the network security. In addition NIDS can detect existence network intrusion. Combination both of NIDS and firewall namely NIPS can detect and deny existence of intrusion. Meanwhile, network security the highly performance depends of the firewall policy rules (Tibbs R.W. and Oakes E.B., 2006 and Suehring S. and Ziegler R.L., 2006). To generate and managing firewall policy rules require a lot of time. An analyst must review all the data of network traffics previously. Therefore it is needed a process of creating firewall rules that reflect the previous network traffics.

CHAPTER 2

LITERATURE REVIEW

This chapter explains the basic concepts and related work that is part of the methodology**.** The basic concept are Intruder, Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Log Files, Firewall and ID3 Algorithm Decision Tree Classifier. All of the concept and related work are presented in this chapter.

2.1 INTRUDER

There are many alternatives to perform intrusion. The intruder will find all information about the target computer system and take advantage of the weaknesses of computer systems. Intrusion can be prevented by always update the system and issues of computer network security (Rafiudin R., 2002).

There are four main categories of intrusion, which are denial-of-service (DoS), probe, user-to-root (U2R) and remote-to-local (R2L) (Khoi-Nguyen T. and Huidong J., 2010; Theodiridis S., 2006 and Ye Q. et al., 2010). Each of these categories represents the generalization of specific attack types. These main categories represent the classification of types of behaviors that can be grouped logically together. For each category, there are multiple attack types and unique to a particular pattern. Table 2.1 shows categories and types of intruder.

Table 2.1 Categories and examples of intruder

CATEGORY	EXAMPLES
DoS	apache2, back, land, mailbomb, Neptune, pod, processtable, smurf, teardrop, upstorm
probe	portsweep, ipsweep, nmap, mscan, saint, satan
U2R	buffer_overflow, httptunnel, loadmodule, perl, ps, rootkit, sqlattack, xterm
R2L	spy, ftp_write, guess_passwd, imap, multihop, named, phf, sendmail, snmpgetattack, snmpguess, warezclient, warezmaster, worm, xlock, xsnoop

2.1.1 Denial-of-Service (DoS)

Denial-of-Service (DoS) attacks is generally known by its attempts to interrupt a service provided as a part of a network (Chandrasekar A. et al., 2009; Dong S.K. et al., 2005; Firewall is it Needed, 2008; Guan X. and Yun-jie L., 2010 and Khoi-Nguyen T. and Huidong J., 2010). An example of a DoS attack is when a service is flooded with requests that it cannot respond effectively "denying" service to any request. The Transport Control Protocol (TCP) with flag SYN (synchronize/start) flood attack is type of attack, wherein SYN connection requests are made that never closed. DoS attacks have crippled websites for extended periods of time. Some have even taken down large portions of the internet, due to the traffic bottlenecks created.

2.1.2 Probe

Probe are the searches for network vulnerabilities to be used in other attacks (Chandrasekar A. et al., 2009; Dong S.K. et al., 2005 and Guan X. and Yun-jie L., 2010). Typically, a network scanned to find servers, and the servers are then scanned for open ports or known vulnerabilities. The signatures of these attacks are usually easy to identify, due to their searching nature. The danger which they impose is their ability to find vulnerabilities that can be leveraged in another attack. The difficulty in network security is the balance between usability and security. If all resources are tightly restricted, then users are limited in the features or applications they can access. If the

security is loose then it become vulnerable toward attacked. Most networks tend to be loose in their security practices. This is why probes are a successful form of intrusion and pose a threat to security.

One example of a probe attack is a “port scan”. This attack uses the approach of incrementally making request to service port on a system. It verifies whether a known service is running on that port, and it learns the attributes of the service. With this knowledge, a more threatening attack can be formed that advantage of vulnerabilities in poorly secured services.

2.1.3 User-to-Root (U2R)

This attack where an intruder exploitation began in systems with normal user account and attempts to abuse the vulnerability in the system to get super user privileged. U2R attack can be the most damaging to system integrity. A U2R attack consists of a user with normal access privileges counterfeiting root level access and full control over the system. Most of these intrusions with buffer overflows in the operating systems that allow a user to gain root access. With root access, the intruder has complete access and control over the machine. Integrity of data and information can be lost or damaged. If intruders gain access to the system as root, it has a greater ability to hide from the intrusion detection.

2.1.4 Remote-to-Local (R2L)

A remote-to-local (R2L) attack allows a remote, non-authorized user to simulate local user privileges on machine (Chandrasekar A. et al., 2009; Dong S.K. et al., 2005 and Guan X. and Yun-jie L., 2010). There are different approaches, including: dictionary-based password and username guessing, attacking vulnerabilities or bugs, and attacking poorly the configured services. Some of these attacks, such as dictionary attacks, are more easily detected than others. As an authorized user on the machine, the intruder can gain the access to private information, disrupt certain services, can corrupt data, or install applications that allow him or her to gain the control of services that can

be used for other malicious behavior. This type of attack can lead to security breaches that allow complete access and control of the machine (user-to-root).

2.2. INTRUSION DETECTION SYSTEM (IDS)

Intrusion detection is detecting actions that attempt to compromise the confidentiality, integrity or availability of resources (Gollman D., 2006; Duanyang Z. et al. 2010; SANS Institute, 2008; Shingo M. et al., 2010 and Weenke L., 2001). When Intrusion detection takes a preventive measure without direct human intervention, then it becomes an Intrusion Prevention System.

Intrusion detection can be performed manually or automatically. Manual intrusion detection is examining log files and then determine network packet is intrusion or not. A system that performs automated intrusion detection is called an Intrusion Detection System (IDS) (Kenneth G.J., 2005; Rafiudin R., 2002 and Winding R., 2006).

Intrusion Detection System the operating systems that allow a user to gain root access. With root access, the intruder has complete access and control over the machine, system is very important in network security. Meanwhile, computer networks continue to expand. IDS need to be able to deal with a large computer network. Therefore, automatic procedures for detecting and responding to intrusion are becoming increasingly essential (Mehmed M.K. and Jozef Z., 2005).

Misuse detection searches for patterns user behavior that match intrusion, which are stored as signatures. These hand-coded of signature are laboriously provided by human experts based on their knowledge of intrusion techniques. If a pattern match is found, then the alarm will appear as a sign. Human security analysts evaluate the alarms to decide what action to take, whether it is shutting down part of the system, alerting the internet service provider about suspicious traffic. An intrusion detection system for a large complex network may produce thousands or millions of alarms per day. Because systems are not static, the signatures need to be updated whenever new software versions arrive or changes in network configuration (Al-Shaer E.S. and Hamed H.H.,

2004). There are two main types of IDS pursuant where data analyzed, which describe in sub chapter 2.2.1 and 2.2.2.

2.2.1 Network-based IDS

Network Intrusion Detection System (NIDS) is an attack of signatures in network traffic. Typically, a network adapter running to monitors and analyzes all network traffic in real time. NIDS is an independent platform that identifies intrusions by examining network traffic and monitors multiple hosts. NIDS gains access to network traffic by connecting to a hub or network switch. An example of a NIDS is software namely *snort* (Gollman D., 2006; Flior E. et al., 2010 and SANS Institute, 2008).

2.2.2 Host-based IDS

Host Intrusion Detection System (HIDS) looks for attack signatures in log files of hosts (Duanyang Z., 2010 and Gollman D., 2006). HIDS are attempts to identify unauthorized and anomalous behavior on a specific device. HIDS generally involves an agent installed on system, monitoring and alerting on local OS and application activity. The installed agent uses a combination of signatures, rules, and heuristics to identify unauthorized activity. The role of a host IDS is passive, only gathering, identifying, logging and alerting of intrusion (SANS Institute, 2008).

There are other ways of finding intrusion attempts. One alternative method is to load software to look for signs of intrusion on the system itself. If a machine has been exploited, often certain system files will be altered. For examples, the password file may be changed, users may be added, system configuration file may be modified, or file permissions might be altered. By looking at changes in these files, there is an intrusion or other unusual activity. There are two classification IDS pursuant how data analyzed, which describe in sub chapter 2.2.3 and 2.2.4.

2.2.3 Misuse Detection

In misuse detection, the IDS analyzed the information it gathered and compare it with large databases of attack signatures. Essentially, the IDS are looks for a specific attack that has already been documented. Learning system from the existing pattern of attack and recognized. This method unable to detect the new attack is which its pattern not yet been known (Duanyang Z., 2010).

2.2.4 Anomaly Detection

In anomaly detection, the system administrator defines the baseline, or normal, state of the network's traffic load, breakdown, protocol, and typical packet size. The anomaly detector monitors network packets to compare their state to the normal baseline and looking for anomalies.

Statistical anomaly detection uses statistical techniques to detect potential intrusions. During operation, a statistical analysis of the data monitored is performed and the deviation from the baseline is measured. If a threshold is exceeded, an alarm is issued. On the other hand, anomaly detection detects just anomalies. Suspicious behavior does not necessarily constitute an intrusion (Gollman D., 2006).

2.2.5 IDS Products

Many IDS systems exist and a lot of confusion because there is little in the way of standards with how they operate. It is difficult to provide a direct comparison between products because terminology, features and functionality. This is because there is no effective comparison can occur. Example of IDS product include *Snort*, *Honeypot* and *Portsentry*.

a. *Snort*

Snort is open source NIDS (Yan Y., 2010). *Snort* capable of performing real-time traffic analysis and packet logging on computer networks. It can perform protocol analysis, searching/matching and can be used to detect a variety of attacks and probes, such as buffer over-flows, stealth port scans, CGI attacks, SMB probes, OS fingerprinting and much mores (Yan Y., 2010).

Snort uses a flexible rules language to describe traffic that it should collect or pass, as well as a detection engine that utilizes a modular plugin architecture. *Snort* has a real-time alerting capability as well, incorporating alerting mechanisms for *syslog* or a specific file user. *Snort* has three primary uses (Yan Y., 2010):

- **Sniffer mode** – to see the packet through the network.
- **Packet logger mode** – for recording all the packet through the network to the analysis at a later.
- **NIDS mode –** *snort* used to detect of attack carried out through computer networks. This mode required NIDS setup of rules that distinguish between normal packets or attack packets.

b. *Honeypot*

Honeypot only provide the security vulnerability of a system, thus providing space for attacked. *Honeypot* is to trap intruder to penetrate computer system. When there are attacks, *honeypot* will be recorded in the log files so the admin can do the next action. *Honeypot* is passive in that they are waiting for someone to attack.

c. *Portsentry*

Portsentry is a software designed to detect port scanning and response actively if any port scanning. The port scanning is a scanning process of services variety applications running on computer servers. Port scanning is the first step before an attack will be undertaken. If there is a machine scan port to servers will actively block the attacker machine.

Portsentry will react in real time by blocking the IP address of the attacker. This was done by using *ipchains* or *ipfwadm* and insert into the file `/etc/host.deny` automatically by the TCP Wrapper. *Portsentry* reports through log files in `/etc/syslog`. The report indicates the system name, time attack, the attacker IP machine IP, protocol, and others. *Snort* and *portsentry* have different action where prevent intrusion, whereas *snort* is only detects the intrusion.

IDS produce has a different way of working and has a different way of prevention. Table 2.2 shows the comparison of work and prevention between *Snort*, *Honeypot a*nd *Portsentry*.

Table 2.2 Comparison IDS products

IDS Product	Work	Prevention
Snort	Intrusion network packet is known based on the rules. When a network packet has a similarity with a rule, then network packet is the intrusion. New attack has a new signature so that rules should always be updated	✔ generate log ✔ passive
Honeypot	Provide the security vulnerability of a system. When there are attacks, *honeypot* will be recorded in the log files.	✔ generate log ✔ passive
Portsentry	Detect machine who is doing port scanning	✔ generate log ✔ active ✔ blocking IP address of intrusion into file /etc/host.deny

2.3 INTRUSION PREVENTION SYSTEM (IPS)

Intrusion Prevention System prevent from intrusion or attacks. IPS work with an IDS, and vendors have combined the two technologies to make an IPS-capable IDS. Two techniques are used to prevent an attack (Rafiudin R., 2002) :

- **Sniping** – Allow the IDS to terminate a suspected attack from through the use of a TCP reset packet or ICMP unreachable message.
- **Shunning** – Allow the IDS to automatically configure router or firewall to deny traffic based on what it has detected and therefore shunning the connection. As IDS become more advanced, this shunning is evolving into a new term, blocking, where an IDS contacts a router or firewall and creates an access control list (ACL) to block the attacking IP.

The IDS can handle sniping. However, shunning requires the assistance of other device. IDS sensors should report back to a central console that, in turn also generates some responses if so configured. Following are some actions that an IDS generate in repose to an attack (Rafiudin R., 2002):

- **Reconfigure firewall/router** – An IDS with a shun enable configure the firewall to filter out the intruders IP address.
- **Send an SNMP trap** – Configure the IDS to send an SNMP trap datagram to a management console.
- **Generate log** – An IDS can log to Windows event log, Syslog server, pager or even send an e-mail.

2.4 LOG FILES

Log files are another critical facet in total security architecture. It is important to create a policy and strategy for dealing system and application. Log files are useful for three reasons (John H.T., et al., 2004):

1. Log files help with troubleshooting system problems and understanding what is happening on the system.
2. Logs serve as an early warning for both system and security events.
3. Logs can be indispensable in reconstructing events, whether determine an intrusion has occurred and are performing the follow-up forensic investigation.

Following some examples from *Snort* log files are shown Figure 2.1

```
[**] INFO - Possible Squid Scan [**]
04/20-14:06:49.953376 192.168.0.33:1040 -> 192.168.0.1:3128
TCP TTL:128 TOS:0x0 ID:393 IpLen:20 DgmLen:48 DF
******S* Seq: 0x60591B9  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
```

Figure 2.1 *Snort* log file

From Figure 2.1, there is effort for the scan of existence of Squid proxy server IP address 192.168.0.1 and port 3128 from workstation IP address 192.168.0.33, port 1040 and protocol is TCP.

```
Feb 17 21:02:13 (none) sshd[14938]: Failed password for albi
from 172.18.64.26 port 3419 ssh2
```

Figure 2.2 *Syslog* log files

From Figure 2.2, someone tries to log in using the 'failed password' from IP address 172.18.64.26 and port 3419 passing ssh service using TCP protocol.

2.5 FIREWALL

A firewall is placed between two or more networks, usually a private network and public networks. Typical examples are the internet which is a zone with no trust and an internal network which is a zone of higher trust must be protected. The term firewall comes from the fact that by segmenting a network into different physical subnetworks, they limited the damage that could spread from one subnet to another just like firedoors or firewalls. Figure 2.3 shows firewall in computer network.

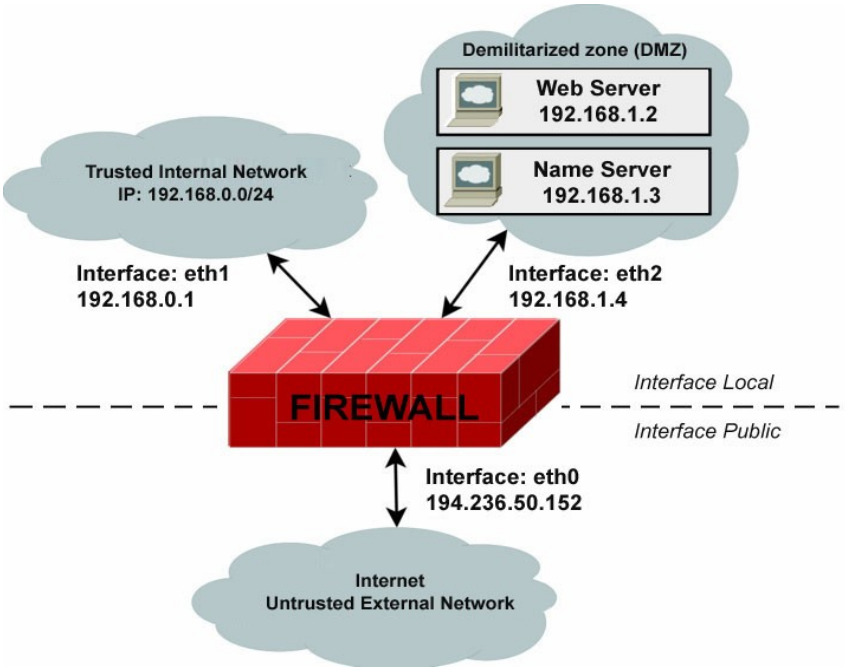

Figure 2.3 Firewall in computer network

Firewall checks every incoming and outgoing network packet to see if it meets the criteria of rules. If it does, firewall will take action to accept, forward, drop or reject the packet network depends the set of rules. Firewalls can filter packets based on source IP address, destination IP address, source port, destination port, protocol and others. Firewalls can filter specific types of network traffic.

The TCP/IP model is older than the Open System Interconnection (OSI) model (Gollman D., 2006; Tibbs R.W. and Oakes E.B., 2006). There are 5 layers in the TCP/IP model. TCP/IP and OSI model is shown in Figure 2.4.